An Efficient Message Collecting and Dissemination Approach for Mobile Crowd Sensing and Computing

Tzu-Chieh Tsai
Department of Computer Science
National Chengchi University
Taipei, Taiwan
e-mail: ttsai@cs.nccu.edu.tw

Hao, Teng
Department of Computer Science
National Chengchi University
Taipei, Taiwan
e-mail: 6105753019@nccu.edu.tw

***Abstract*—Mobile Crowd Sensing and Computing (MCSC) is an emerging technology along with the popularity of mobile devices. We utilize the concept of Delay Tolerant Networks (DTNs) and edge/fog computing to support the message collection and dissemination for the MCSC. The biggest challenge here is to design an efficient routing method to deliver messages for both "upload" (data collection to edge nodes) and "download" (data dissemination to nodes that interest) paths. We assume that the mobile crowd nodes will like to exchange data in a DTN manner based on opportunistic transmission in order to save energy and data transmission cost. We design a probability-based algorithm to upload data carried by normal mobile nodes to the edge nodes. Then, we use cosine similarity to relay specific message of attributes to users who have high interest to receive the message. We simulate the algorithm with the National Chengchi University (NCCU) real trace data of campus students, and compare it with other traditional DTN routing algorithms. The performance evaluations show the improvement of message delivery ratio and decreasing latency and transmission overhead.**

Keywords- Delay Tolerant Network; Mobile Crowd Sensing and Computing; Opportunistic Mobile Networks; Personal Interests; Trace Data

I. INTRODUCTION

The popularity of mobile phones has been growing dramatically recently. Each mobile phone is equipped with many sensors, varieties of wireless communication interfaces (WiFi, Bluetooth, 3G/4G) and sufficient storage space. Therefore, there is a new category of applications arising, namely, Mobile Crowd Sensing and Computing (MCSC) [1]. Unlike traditional wireless sensor networks, MCSC does not require a large number of sensors pre-installed. The mobile devices can cooperate, collect the interested information and exchange with each other. To increase the incentive of the cooperation, the common interest or social relationships may be considered because the mobiles are carried by human beings. Thus, MCSC can be seen as a good way to solve the problem utilizing the power of the human participation.

Mobile Crowd Sensing and Computing has a wide range of applications, such as temperature [2] and air quality detection in the city [3], restaurant recommendations [4] and so on. With the concept of MCSC, there are still many problems need to be solved. The computing power of mobile devices is usually limited or the required data for computation is large. The sensed data from the mobile crowd should be collected into some place such that is suitable for computing. On the other way, the computing results are somehow needed to deliver the interested users who are not restricted to a precise destination. Therefore, both an efficient data collection approach and a message dissemination method are needed to develop.

Nowadays, people can get or dissemination message by social networks, such as Facebook, Twitter. However, the data sensed by mobile crowd are usually to be processed first, and then become usable information to be sent to people who may interest the message.

Previous studies tried to solve the influence maximization problem [5] in the online social network [6][7][8], or how to do trace data processing and data exploration effectively [9]. However, most users are free to participate in MCSC environment [10]. Furthermore, the activity of MCSC should be done on the background and as transparently to users as possible. So there is not enough incentive for users to upload sensor data using their own mobile network with no cost. In addition, the hardware resources and energy of mobile devices are also quite limited. So we utilize the concept the DTN and edge computing to support the MCSC. How to keep data transmission as efficient as possible and save network resources are the key issue for MCSC applications.

The rest of this paper is organized as follows: Section II introduces the motivation using our MCSC scenario as an example. Section III addresses some important related works. Section IV will explain our approach including that the MCSC process is divided into two phases: "upload" and "download" for message collecting and dissemination. Section V validates and evaluates the system performances of our proposed scheme. Section VI concludes this paper.

II. MOTIVATION

Let us think about the following campus scenario as an MCSC example. There are all kinds of messages being spread on a campus. Students carry their mobile devices and move around the campus for attending classes or go to library or restaurant, etc. In this situation, all the students who carry mobile devices on campus are assumed to the mobile crowd nodes. They can generate or collect messages, and receive and transmit certain kinds of messages when they encounter each other. We assume the messages have relevant interest attributes: sports, arts, or social, etc. Students who carry their

 ISBN: 978-1-61208-739-9

message may leave or post the message to the building(s) which we assume to be the edge node(s) for further computing purpose. The edge node(s) gather enough necessary information and process them, and then disseminate the results to the interested people. As in Figure 1, the mobile devices can be the helpers as the routing roles of message relay for collecting and disseminating messages.

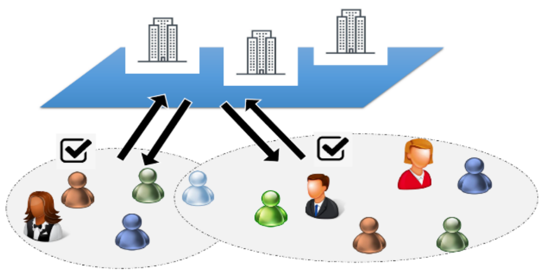

Figure 1. MCSC in our situation.

The facts here impacting the message delivery are the mobility of the students (i.e., the opportunity of encountering somebody to exchange message), and the interest attributes of the message itself and the students (i.e., the willing to carry for relaying the message). There should be a trade-off between message copies and communication overheads. Since students have their hobbies and their class schedules, the message delivery will somehow form as an opportunistic mobile social networks. Our goal of the research will be developing an efficient method for "upload" (message collecting to buildings (edge nodes)) and "download" (computing result dissemination) in this MCSC scenario. The features of MCSC are that the messages collected should be processed or computed first, and then the results could be delivered to those who are interested, but not sure which nodes are destinations in advance as traditional routing.

III. RELATED WORKS

Before further illustrating our approach, some important related works are introduced in this section.

A. *Delay Tolerant Network (DTN)*

DTN is a special network transmission concept. Compared with traditional network architectures, there may be situations such as intermittent disconnection and unable to access Internet. DTN allows nodes to store-and-carry messages, and transmit/forward the message when they get access or encounter somebody else later on to further relay. DTN only needs peer-to-peer communication with the help of infrastructure. So, some applications are suitable for using the techniques, such as disaster response networks, and vehicle-to-vehicle networks.

Since Messages can be passed between nodes and nodes by "store, carry and forward", the decision to forward which messages to the nodes they encounter become the key issues for efficiency of delivery ratio and overheads. How to design an appropriate routing method for message transmission with optimal efficiency is an issue for DTN.

B. *Edge Computing*

Edge Computing refers to the processing and operation of data more locally than cloud computing. It helps message moving closer to the data source so as to shorten the delay of network transmission, as well as to obtain the wisdom of data analysis faster.

The network concept is proposed by Cisco. Compared with traditional Internet architecture, fog computing uses layered, local processing distributed network packet transmission to calculate computing requirements. Each fog is directly linked to the local device on the local side. It is responsible for collecting sensor data, raw data and doing preliminary processing. One fog can communicate with other fogs. In addition, it uploads to the cloud so as to do the best calculation, and do regular information update. The current technology has been able to be applied in the temporary system in the smart grid and some houses.

C. *Social Trace Data File*

We need a realistic social trace data file for MCSC simulation. Social trace data include the mobility trace of the nodes and the social relationship and personal interest for the users who carry the mobile sensors. Even the attributes of the building they visited are also described in the data file.

In the DTN environment, the main purpose of social trace data file is including:

1) To simulate the movement history track of the users.

2) To make the forward policy based on the personal data

Our previous research results have completed a data file called NCCU Trace File [19][20] which includes personal mobility trace for two weeks and personal interest profiles, and already used it for evaluating our developed methods in many scenarios.

IV. OUR APPROACH

Due to the features of MCSC, we will divide the MCSC process as two phases: "upload" and "download" as in Figure 2. Nodes will upload the message to any one of the edge nodes, and we assume some edge to be the designated node to compute the collected sensed data for some specific purpose, and to then send the result messages to the users who are interested in. In the upload phase, the message destination can be any one of the edges nodes, i.e., anycast. In the download phase, it is one-to-many transmission, however, we cannot know in advances who will be the destination until the message encounters the nodes to compare the interest attributes.

A. *Model*

Anypath routing means the final message destination can be any one of the candidate destinations.

In our model, the destinations are buildings which we assume to be fixed edge nodes. Mobile crowd may encounter one of the edge nodes directly or encounter other mobile node before visiting buildings. In the former case, the message that mobile crowd carried can be directly forwarded to the edge node (anycast destination). In the latter case, the node should

 ISBN: 978-1-61208-739-9

decide whether forward the message to the encountered node will have a better chance to faster relay message to destinations.

Since node encountering is based on “opportunity”, and is not certain to meet the “right” person in the near future. We use the concept of most appropriate “forwarding set” to estimate the “cost” for relaying if we forward the message to the encountered node. That is, we determine the appropriate forwarding set by calculating the cost of the transmission between the node and the edge node set. The details will be described in the following section.

According to the above, our model has many nodes (many users in the trace data), and an edge node set (buildings in the trace data). We use the delay-tolerant network technology to transfer the message. And we use anypath routing algorithm to select appropriate nodes to become relay nodes for “upload”. Messages are carried and uploaded by each node to the edge for pre-processing, and then be downloaded to the nodes that need the messages (as in Figure 2).

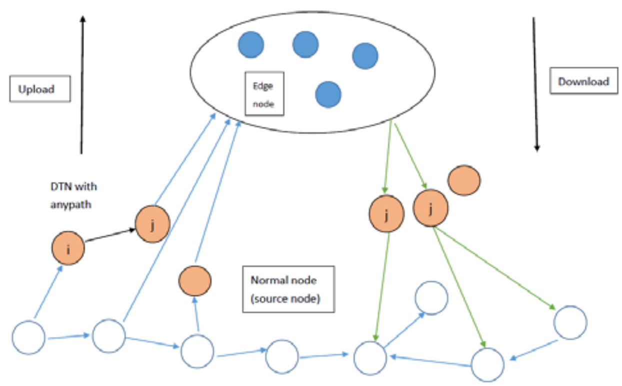

Figure 2. Our approach model.

B. Cost Probability

Due to easy implementation and low maintenance cost, Bellman-Ford-like algorithm is adopted to calculate the cost from node *i* to any destination through the forwarding set. We define the link cost C_{ij} as the reciprocal of the encounter probability between node *i* and *j* (P_{ij}). The forward set J for node *i* will be the future most likely encountering nodes to be forwarded messages to any one of destination set E. Thus, the anypath routing cost for node i to E:

$$C_{iE} = C_{iJ} + C_{JE}$$

$$P_{ij} = \frac{i\ encounter\ j}{\text{i encounter all other nodes}}$$

$$C_{iJ} = \frac{1}{P_{iJ}} = \frac{1}{1 - \prod_{j \in J}(1 - P_{ij})}$$

C_{iJ} is the cost of reciprocal of the probability that the node *i* meets at least one node in the relay set J. Message can be transferred to any node j in Set J to help relay. The cost for Set J to any edge node e_x, C_{Je_x} is:

$$C_{Je_x} = \sum_{j \in J} w_{ij} C_{je_x}$$

where, C_{je_x} represents the path cost for node j to edge node e_x, and w_{ij} the normalized probability that node *i* meets the specific node *j* in the relay node set J.

$$w_{ij} = \frac{P_{ij} \prod_{k=1}^{j-1}(1 - P_{ik})}{1 - \prod_{j \in J}(1 - P_{ij})}, \ \sum_{j \in J} w_{ij} = 1$$

Finally, the C_{JE} can be derived as the following:

$$C_{JE} = \min\{C_{Je_x}\}$$

C_{JE} indicates the cost that relay node set J reach at least one of an edge node e_x in an edge node set E. Edge nodes will do the message preprocessing as mentioned before, as long as enough message are uploaded and concentrated to a designated node to computing results.

Whenever, node *i* encounters node *j*, the C_{iE} and C_{jE} will be compared, then make the forwarding decision based on Bellman-Ford shortest path algorithm.

C. Relay Node Set

We use the above calculation to select the appropriate nodes to form the relay node set. The basic idea is to choose relay nodes with the highest probability to future encounter and lowest cost to any edge nodes. Actually, the relay node set should use appropriate size to estimate the anypath cost better. Our solution set the size of relay node set to be three with minimum cost to edges. (as shown in Figure 3)

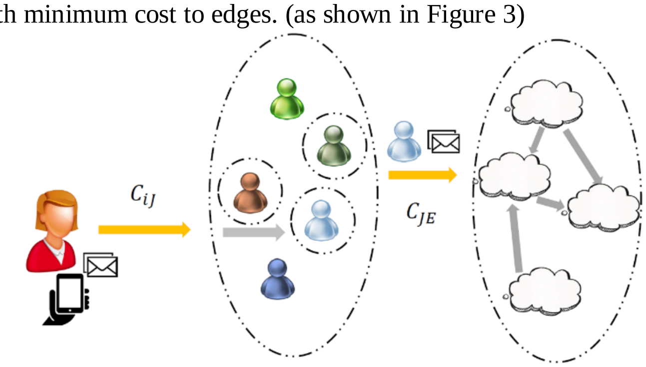

Figure 3. Selection of Relay Nodes.

D. Cosine Similarity

In the download phase, the result message will be delivered to those who are interested. To decide whether the user interests the message or not, we use “cosine similarity”. There are many methods in computer science that can help us measure vector similarity. Vector similarity can help us classify different categories. Classic vector similarity includes Euclidean Distance, Cosine Similarity, Hamming distance, or Jaccard similarity, etc. Different similarity methods are used at different cases due to their characteristics. For proof of concept only, we use the cosine similarity for simplicity. Since each message has its interest attributes and every student has his/her interest profile, we define the interest *I* between node *j* and message *m* as the following:

$$I(j,m) = \frac{\vec{j}, \vec{m}}{\|\vec{j}\| \cdot \|\vec{m}\|} = \frac{\sum_{k=1}^{n} j_k \times m_k}{\sqrt{\sum_{k=1}^{n}(j_k)^2 \times \sum_{k=1}^{n}(m_k)^2}}, \ k < n$$

ISBN: 978-1-61208-739-9

$\vec{j}$ is the interest vector of node j, and $\vec{m}$ is the interest vector of message m. The relay node j has its own interest attributes same as message attributes(i.e. m_k and j_k includes sports, art, reading, service, social like we mentioned before). The message sensed/generated by the initial node also has its own interest attributes. We compare the message and the relay node by each other's interests, and use cosine similarity to set up an association. We also set a similarity threshold. If the cosine similarity of message and node is greater than the similarity threshold, we can determine that the node is interested in this message enough. We will relay the message to this relay node, and this node will also become the destination of this message in the download phase.

E. Influence Gain

We calculate whether this node is appropriate to help us relay based on the history record of this node. Our NCCU trace data has two-week data. If the data of first week is calculated, the second week's data will be used as a reference. On the contrary, if the data of second week is used, the first week's data will be used as a reference. The reason is that our NCCU trace uses campus data, so we can assume that the student's mobile and interactive records may be related to their weekly schedule whenever weekdays or holidays.

We consider that node i with message m encounters node j: If the node j does not have enough interest in the message m, we then determine whether node j might meet with high possibility other nodes who are interested. If yes, then the message could be relayed to it. To do this, we introduce the influence gain that calculates the normalized "interest inherited" from those who might be met in the future. The inherited interest is sum of the interest of those who met on the same day of the other week of the record. Thus, influence gain (IG) of nod i is:

$$IG(i) = \sum_{j \in J_D} P_{ij} * \text{normalized\_inherited\_interest}(j)$$

,where J_D is the forward set of node i on the same day D.

If $I(IG(j), m) > I(IG(i), m)$, and $I(IG(j), m) >$ Threshold, then the message m will be forwarded to node j.

V. SIMULATION

We use The One Simulator, a network simulator based on the action mass perception network developed by the Norwegian Nokia research engineer. The "ONE" is the acronym for The Opportunistic Network Environment simulator, which is an open source development tool available on GitHub.

The One is made with Java and implemented in a GUI interface. It can fully simulate the routing results of network nodes over a specific time period. Among the ONE simulator, we have built-in multi-node transmission methods including one-to-one, many-to-many, one-to-many and many-to-one. There are also built-in Epidemic, Spray & Wait, Prophet and other classical DTN routing methods for users to do simulation experiments. Since the MCSC in our cases is different from traditional DTN routing, we can only compare with epidemic which is similar to flooding as the baseline (anycast in the upload phase, and no information about destinations in the download phase at the time of route computation). Existing related works cannot be directly applied to these MCSC cases.

For the sake of fair comparisons, in our experiments, we modified the epidemic routing to be fit in our scenario for both the upload and download phases. Notice that the message received in the upload phase should exclude the duplicate; however, in the download phase there are possible many destinations for each message.

A. Latency

As the latency is concerned, compared with the epidemic, our method achieves better performance in most days of the week as depicted in Figure 4. This is because our method forwards the message to appropriate nodes without overloading the nodes' buffer. The epidemic method might quickly fill the buffer and takes time to "digest" the message, thus, incurring more latency.

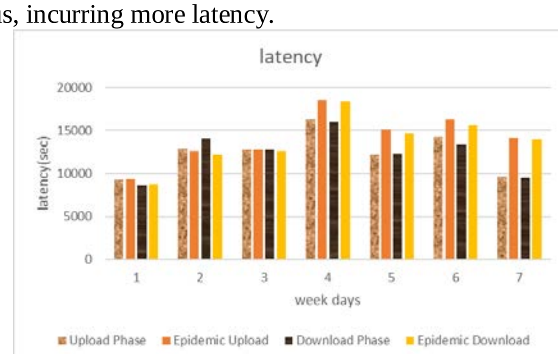

Figure 4. Latency simulation.

B. Hop Count

Average Hop count for the path to destinations is also one of the important factors in DTN consideration. Our method outperforms epidemic (in Figure 5). This again confirms our method picks more suitable relay node(s) to help relay messages efficiently. Especially in download, the hop count is significantly reduced.

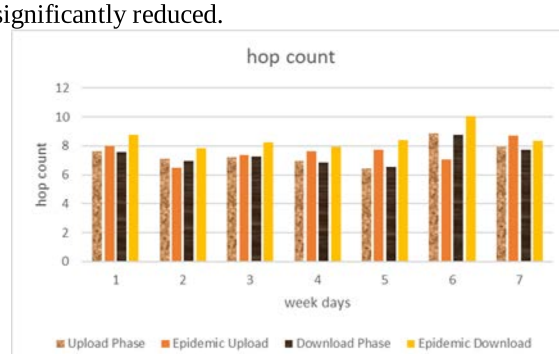

Figure 5. Hop count simulation.

C. Delivery Ratio

We calculate the delivery ratio from the process of generating the message to the end of the arrival. It is expressed as the proportion of number of messages successfully delivered to the number of all messages transmitted. The formula can be expressed as:

$$\text{Delivery Ratio} = \frac{\text{Number of successful msg to Destination}}{\text{Number of msg be sent}}$$

Because of efficient selection of relay node set, the number of messages that can reach the destination is much higher. So, in most cases there will be a higher delivery ratio whether it is upload or download phase in our approach model.(Figure 6)

Figure 6. Delivery ratio simulation.

D. Overhead ratio

Finally, the overhead ratio is considered. The overhead ratio here refers to the proportion of messages that are wasted among all the transmitted messages. The formula can be expressed as:

$$\text{Overhead Ratio} = \frac{\sum_{m_i}^{M} \frac{\text{Relayed} - \text{DestinationRelayed}}{\text{DestinationRelayed}}}{\text{Total Message Number}}, \forall m_i \in M$$

In the process of Epidemic, because the probability of random transmission is very high, the amount of information received by each node is likely to cause buffer problems. Because our message is composed of interests options, so it is easy for messages to be truly transmitted to destination with fully interests. (Figure 7)

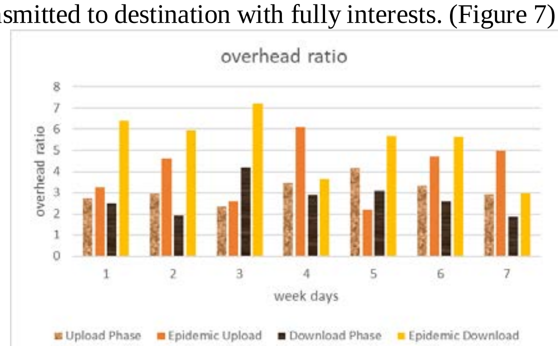

Figure 7. Overhead simulation.

E. Weekdays and Weekend

Our trace data is mainly to record the walking and interaction records of campus students to help us deliver messages in two weeks. Therefore, our data and simulation results may have more obvious differences between weekdays and weekends compared with other general environment or situation.

According to the results (Figure8-11), Weekdays have fewer hop counts than weekends because there are more choices in the case of a large number of students. Especially our approach still has a good overhead ratio. In the epidemic, there is a higher overhead in the weekdays because of the number of transmission choices.

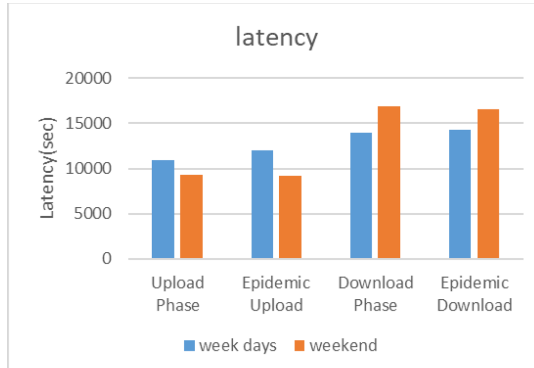

Figure 8. Latency of week days and weekend.

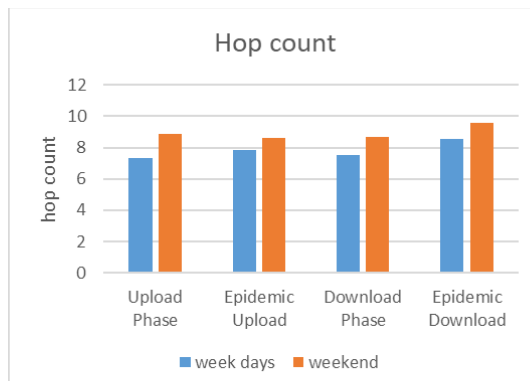

Figure 9. Hop count of week days and weekend.

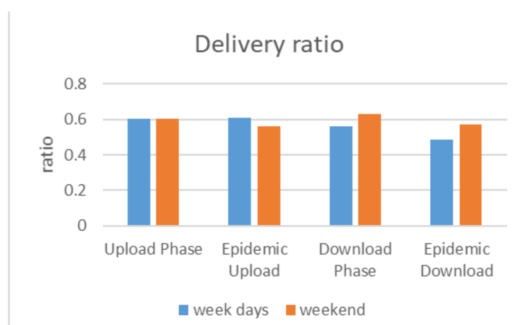

Figure 10. Delivery ratio of week days and weekend.

 ISBN: 978-1-61208-739-9

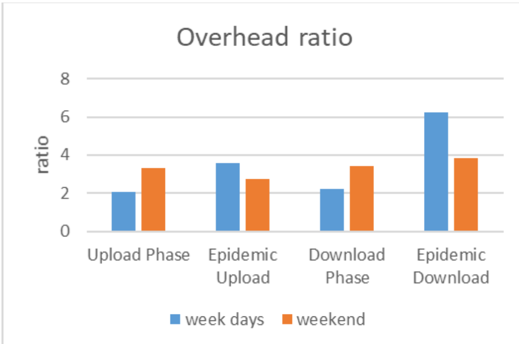

Figure11. Overhead of week days and weekend.

VI. Conclusion and Future Works

This paper presents the design of an efficient approach with upload and download phase for MCSC applications. The developed approach tries to transfer the specific message with interests based content that everyone carries to the people who really need it. We use the concept of Mobile Crowd Sensing and Computing to bring the power of the masses to the limit by the mobile device in the user's hand as the node. We use the NCCU trace data record as the benchmark to calculate the probability of encounter and reaching the edge node in the upload phase. According to the calculation result, the message will be uploaded to the main processing edge. The edges will help compute messages. The result messages will be ready to be transmitted for the download phase. We propose the inherited interest from those who are possibly met in the future to determine the relay nodes. The performance evaluations show the improvement of message delivery ratio and decreasing latency and transmission overhead.

We might consider the latency constraints for the message into our routing method in the future. Although the delay in the DTNs is not very stringent, it still needs the lifetime limit in some cases or to avoid bandwidth wastage. Another direction is the buffer size. Since DTNs use the store-carry-and-forward approach to relay messages, the mobile nodes may store too many messages and over utilize their computing capabilities. A real mobile APP can be practiced to ensure the merits of this research.

References

[1] H. Ma, D. Zhao, and P. Yuan, “Opportunities in mobile crowd sensing”, IEEE Communications Magazine, August, 2014.

[2] M. Mun, et al, “PEIR, the personal environmental impact report, as a platform for participatory sensing systems research”, Proceedings of the 7th international conference on Mobile systems, applications, and services. ACM, 2009. pp.55-68.

[3] C. Xiang, et al, “Passfit: Participatory sensing and filtering for identifying truthful urban pollution sources”, Sensors Journal, IEEE, 2013.

[4] S. Gaonkar, J. Li, R. Choudhury, L. Cox, and A. Schmidt, “Micro-blog: sharing and querying content through mobile phones and social participation”, In: Proceedings of the 6th international conference on Mobile systems, applications, and services. ACM, 2008. pp. 174-186.

[5] D. Kempe, J. Kleinberg, and E´. Tardos, “Maximizing the spread of influence through a social network”, ACM SIGKDD, 2003, pp. 137–146.

[6] W. Chen, Y. Wang, and S. Yang, “Efficient influence maximization in social networks”, ACM SIGKDD, 2009, pp. 199–208.

[7] M. Han, M. Yan, Z. Cai, and Y. Li, “An exploration of broader influence maximization in timeliness networks with opportunistic selection”, Journal of Network and Computer Applications, 2016.

[8] T.Shi, J.Wan, S.Cheng, Z.Cai, Y.Li, and J.Li, “Time-bounded positive influence in social networks”, in IIKI, 2015.

[9] B. Guo, et al, “Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm”, ACM Computing Surveys (CSUR), 2015, 48.1: 7

[10] T. Higuchi, H. Yamaguchi, T. Higashino, and M. Takai, “A neighbor collaboration mechanism for mobile crowd sensing in opportunistic networks”, Communications (ICC), 2014 IEEE International Conference on. IEEE, 2014. pp.42-47.

[11] N. Eagle and A. Pentland, “Reality mining: sensing complex social systems”, Personal and Ubiquitous Computing, Vol 10(4):255–268, May 2006.

[12] P. Hui, “People are the network: experimental design and evaluation of social-based forwarding algorithms”, Ph.D. dissertation, UCAM-CL-TR-713. University of Cambridge, Comp.Lab., 2008

[13] R. Marin, C. Dobre, and F. Xhafa, “Exploring Predictability in Mobile Interaction”, EIDWT. 2012.

[14] J. Cai, M. Yan, and Y. Li, “Using crowdsourced data in location-based social networks to explore influence maximization”, The 35th Annual IEEE International Conference on Computer Communications (INFOCOM 2016). 2016.

[15] J. Qin, et al, “Post: Exploiting dynamic sociality for mobile advertising in vehicular networks”, IEEE Transactions on Parallel and Distributed Systems, 2016.

[16] R. Mayer, H. Gupta, E. Saurez and U. Ramachandran, “The Fog Makes Sense: Enabling Social Sensing Services With Limited Internet Connectivity”, ACM New York, 2017.

[17] K. Dolui, and S. Datta, “Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing”, IEEE Global Internet of Things Summit (GIoTS), 2017.

[18] T. Tsai, and H. Chan, “NCCU Trace: social-network-aware mobility trace”, Communications Magazine, IEEE, 2015.

[19] T. Tsai, H. Chan, C. Han, and P. Chen, “A Social Behavior Based Interest-Message Dissemination Approach in Delay Tolerant Networks”, International Conference on Future Network Systems and Security. Springer International Publishing, 2016. pp. 62-80.

[20] R. Laufer, H. Dubois-Ferriere ,and L. Kleinrock, “Multirate Anypath Routing in Wireless Mesh Networks”, IEEE INFOCOM, 2009.

[21] H. Li, T. Li, W. Wang, and Y. Wang, "Dynamic Participant Selection for Large-Scale Mobile Crowd Sensing," in IEEE Transactions on Mobile Computing.

[22] Z. Zhou, H. Liao, B. Gu, K. M. S. Huq, S. Mumtaz, and J. Rodriguez, “Robust Mobile Crowd Sensing: When Deep Learning Meets Edge Computing”, IEEE Network Volume: 32, Issue: 4, July/August 2018.

